Assignment 1

//

// main.cpp

// Polynomial

//

// Created by sahana cs on 1/30/18.

// Copyright © 2018 sahana cs. All rights reserved.

//

#include <iostream>

using namespace std;

class term {

public:

int power;

int coef;

term \* next;

term() { next = nullptr;}

term(int c , int p){ power = p ; coef = c ; next = nullptr; }

};

class polynomial{

public:

int num\_terms;

term \* head;

polynomial(){ num\_terms = 0; head = nullptr; }

void add\_term(int c, int p);

polynomial operator+(polynomial p);

polynomial operator\*(polynomial p);

void print\_polynomial();

};

void polynomial::print\_polynomial(){

cout<<endl;

term \* p = head;

while (p != nullptr) {

cout << p->coef << "," << p->power << " -> ";

p = p->next;

}

}

void polynomial::add\_term(int c, int p){

term \* add = new term(c,p);

if(head == NULL)

head = add;

else if (head->next == nullptr){

if(head->power >= add->power){

if(head->power == add->power){

head->coef = head->coef + add->coef;

if(head->coef == 0){

head = NULL;

delete add;

}

}

else{

head->next = add;

}

}

else {

add->next = head;

head = add;

}

}

else{

if(add->power >= head->power){

if(add->power == head->power)

{

head->coef = add->coef + head->coef;

if(head->coef == 0){

delete add;

head = head->next;

}

return;

}

else{

add->next = head;

head = add;

return;

}

}

else {

term \*cur = head->next;

term \*prev = head;

while(cur!=nullptr){

if(add->power >= cur->power){

if(add->power == cur->power){

cur->coef = cur->coef+add->coef;

if(cur->coef == 0){

delete add;

prev->next = nullptr;

}

return;

}

else{

prev->next = add;

add->next = cur;

return;

}

}

else

{

prev = cur;

if(cur->next == nullptr){

prev->next = add;

return;

}

cur = cur->next;

}

}

}

}

}

polynomial polynomial::operator+(polynomial p){

polynomial \*result = new polynomial();

term \*t1 = head;

term \*t2 = p.head;

while(t1 != nullptr || t2 !=nullptr)

{

if(t1!= nullptr && t2 !=nullptr){

if(t1->power == t2->power){

int sum = t1->coef + t2->coef;

result->add\_term(sum,t1->power);

t1 = t1->next;

t2 = t2->next;

}

else{

if(t1->power > t2->power){

result->add\_term(t1->coef, t1->power);

t1 = t1->next;

}

else{

result->add\_term(t2->coef, t2->power);

t2 = t2->next;

}

}

}

else{

if(t1 == nullptr){

while(t2 != nullptr){

result->add\_term(t2->coef, t2->power);

t2 = t2->next;

}

}

else{

while (t1 !=nullptr) {

result->add\_term(t1->coef, t1->power);

t1 = t1->next;

}

}

}

}

return \*result;

}

polynomial polynomial::operator\*(polynomial p){

polynomial \*result = new polynomial();

term \*t1 = head;

while (t1 != nullptr){

term \*t2 = p.head;

while(t2 !=nullptr){

int product = t1->coef \* t2->coef;

int power = t1->power + t2->power;

result->add\_term(product, power);

t2 = t2->next;

}

t1 = t1->next;

}

return \*result;

}

int main() {

polynomial p1,p2,p3,p4;

int num\_t;

cout<<"enter number of terms" << endl;

cin>>num\_t;

cout<<"Enter all terms" << endl;

for(int i =0; i< num\_t ; i++){

int c,p;

cin >> c >> p ;

if( c!= 0)

p1.add\_term(c,p);

}

p1.print\_polynomial();

cout << endl;

cout<<"enter number of terms" << endl;

cin>>num\_t;

cout<<"Enter all terms" << endl;

for(int i =0; i< num\_t ; i++){

int c,p;

cin >> c >> p ;

if(c!=0)

p2.add\_term(c,p);

}

p2.print\_polynomial();

cout << endl;

cout<<"Addition:";

p3 = p1 + p2;

p3.print\_polynomial();

cout<<endl<<"multiplication:";

p4 = p1 \* p2;

p4.print\_polynomial();

cout<<endl;

getchar();

getchar();

return 0;

}